

Your name: _____

Your student number: _____

Your professor's name: _____

Your section number: _____

**University of Saskatchewan
Department of Computer Science
CMPT 370**

**Final Examination
December 10, 2001**

Time Limit: 3 hours

Total Marks: 100

This is a closed book exam. Please write your answers legibly in the space provided on the examination paper. If you need more space, use the back of the page, but you should note that the space provided corresponds approximately to the length of answer desired. Rough work can be done in the answer booklets. Be sure to budget your time appropriately so you can answer all questions (there are just under 2 minutes per mark).

1. Very, very short answers: [5 marks, 1 mark each]

- 1.1 In Canada, a person can be certified as software professional using this designation:
- 1.2 Responsibility is assigned to an object using:
- 1.3 If every member of a class A must also be a member of a subclass of A, A is a:
- 1.4 The analysis of the world in which the software is to be applied is captured in a:
- 1.5 The software processes of a company can be classified according to the:

2. Very short answers: [14 marks, 2 marks each]

- 2.1 What is the most significant difference between the waterfall model and the UP?
- 2.2 What is whistle blowing?
- 2.3 Why is it important to establish a system boundary early in a project?
- 2.4 What is meant by “timeboxing” an iteration when developing software?
- 2.5 Why is reverse engineering a possible threat to intellectual property?
- 2.6 What is the distinction between debugging and testing?
- 2.7 How are architectural decisions documented?

3. Short answers: [25 marks, 5 marks each]

- 3.1 For each definition in the second list choose the most appropriate term from the first list. No term in the first list should be used more than once. Write the term beneath the appropriate definition. [5 marks]

First List, terms:

system sequence diagram	verification
service factory	responsibility
validation	cardinality
coupling	use case
cohesion	role

Second List, definitions:

- testing whether the system been built right
- captures the interactions of external actors and the system
- a dependency between elements
- a named end of an association to indicate its purpose
- a service provided by an object or a subsystem

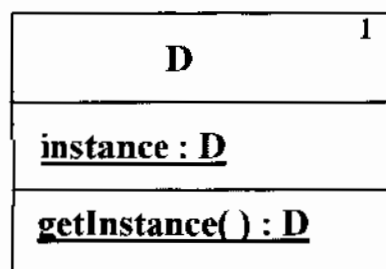
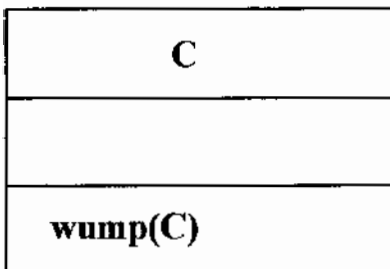
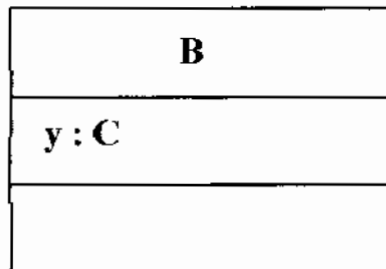
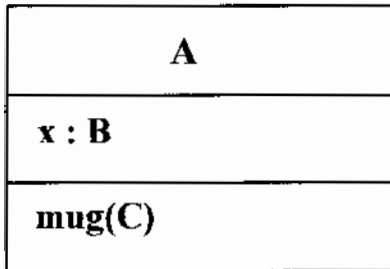
- 3.2 Many design patterns mutually support one another. What other design patterns support “protected variations”? Explain. [5 marks]
- 3.3 In the presentation by the software firm EDS a commercial was shown in which workers were still building an airplane while it was already in service flying passengers. The commercial concluded that “in a sense, this is what we [EDS] do”. Why is building an airplane in flight an analogy for software design? What does this imply about the differences between software engineering and the engineering of physical artifacts such as an airplane? [5 marks]

3.4 Briefly describe the four phases of the unified process. In your discussion, describe the main artifacts that may be created or refined in each phase. [5 marks]

3.5 The unified process suggests six (6) views of software architecture. List and describe four (4) of these views. [5 marks]

4. Visibility question: [6 marks]

Consider the following partial class diagram:



- 4.1 For each class in the diagram, indicate which other classes are visible and what kind of visibility it is. [4 marks]

Visible from A:

Visible from B:

Visible from C:

Visible from D:

- 4.2 Flesh out the class diagram by drawing in the known navigability links between classes and any known dependency relationships. [2 marks]

5. Design patterns question: [10 marks, 2 marks each]

In each of the following situations, indicate which design pattern has been used or is likely to be useful in handling the situation. Choose the *most appropriate* design pattern from the list of design patterns below. Choose *only* from this list and choose *only one* for each situation. No pattern should be chosen more than once. There should thus be 5 unused patterns! Justify your choice in a single sentence explaining *why* that particular design pattern is most appropriate.

List of Design Patterns (choose only from this list):

singleton	observer
state	composite
layers	strategy
factory	pure fabrication
high cohesion	proxy

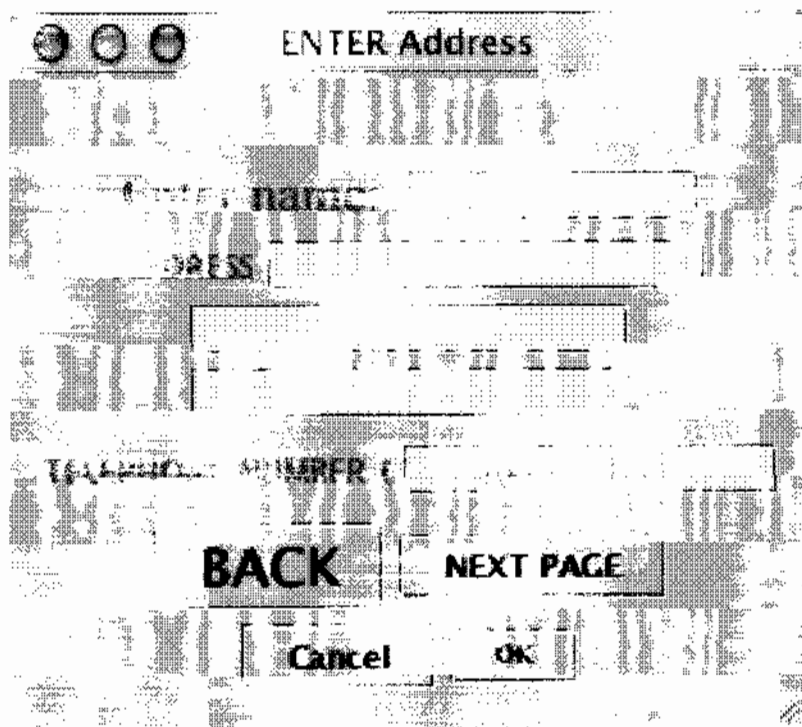
System Design Situations:

- 5.1 The police services of Bigville need to access data about known criminals from the information systems of many other police services. The designers of the Bigville Criminal Information System (BCIS) have decided to have objects in their system corresponding to each of these other information systems to make sure any BCIS data request conforms to the particular security requirements of that other information system. Which design pattern has been used?
- 5.2 The Alpha-Beta company makes many different types of complex mechanical devices. Each device consists of many parts. Each kind of part can be in many types of devices. Each device and part is modeled as a software object in Alpha-Beta's inventory control system. When the price of a part changes, the price of the devices that use the part must also change. Which design pattern might help in keeping the prices up to date?
- 5.3 The local bookstore is building a system to support its frequent buyer program. Based on the number of purchases made, each customer is categorized as a light, medium, heavy, super, or colossal buyer. When a customer makes a new purchase, special deals and discounts are applied depending on the customer's buyer category. Which design pattern would help the system apply these differential deals and discounts?

- 5.4 SaskRiver.com is an on-line service selling Saskatchewan crafts through the internet. In the software system supporting this service there is an object representing each craftsman. Each such object has complex relationships with other objects in the system, depending on the particular kind of craft(s) made by the craftsman. Which design pattern would help to incorporate new craftspeople into the system?
- 5.5 In a large financial system many reports are produced. It has been decided that all the reporting functions will be put together into one package in order to insulate the rest of the system from future changes to the report formats. Which design pattern has been used?

6. Graphical User Interface Question: [5 marks]

The following dialog box is an example of a poor user interface design. Circle the portions of the dialog box that violate the guidelines to good interface design. Annotate each circle with the name of the guideline violated, i.e. with one of the following terms: **visual consistency**, **visual organization**, **legibility**, or **readability**.



7. Design Question [35 marks]

In keeping with recent directives to enhance revenue opportunities, the University of Saskatchewan is about to launch a discount airline called SaskAir. SaskAir will have flights to over a dozen cities throughout the length and breadth of Saskatchewan, with numerous daily arrival and departure times. To keep costs down, SaskAir will use a single type of aircraft in its fleet and there will be a single class of service and thus a single fare for each flight.

You have been asked to design the SaskAir Reservation System (SRS) to track passenger reservations. Over the internet, the customer will be able to make a reservation, cancel a reservation, purchase a ticket and browse available flights. A SaskAir clerk will maintain the flight schedules and fares and be able to check to see how full a flight is and to create a passenger list. A SaskAir manager will be able to see how much revenue the airline is generating for each of the flights.

All ticket purchases are handled by a third party system called UPAY. Your SaskAir reservation system needs to interface with the UPAY system, providing reservation information and receiving a confirmation whether the payment transaction was successful or not. Upon a successful reservation and payment transaction, the SaskAir reservation system will email the itinerary and a confirmation number to the passenger. The passenger will use the confirmation number instead of a ticket when traveling with SaskAir.

The following questions apply to the SaskAir reservation system.

- 7.1 Identify and list the main conceptual classes for the SaskAir reservation system.
[3 marks]

7.2 Identify and list the actors for the SaskAir reservation system. [2 marks]

7.3 Draw a use case diagram for the SaskAir reservation system. [5 marks]

- 7.4 Draw a design class diagram for the SaskAir reservation system. Show associations, significant attributes, significant methods, and multiplicity expressions. [7 marks]
- 7.5 Write a contract for the **bookFlight** operation in which a customer checks for available flights between two cities, selects a flight for a particular date, makes a reservation, pays for their ticket, and receives a confirmation. [4 marks]

7.6 Draw a collaboration diagram (or sequence diagram) for the **bookFlight** operation described in 7.5. [10 marks]

7.7 Annotate the interaction diagram above with the main design patterns used. [4 marks]